FINAL REPORT: PRACTICE SCHOOL 1

Single Axis Solar Tracker

BY:

Abhinav Srivastava[2020A3B50708G]
Atharva Uday Deshmukh[2020A8PS1799P]
Saksham Subhash Yadav[2020A8PS2156H]

AT:

Prama Instruments Pvt Ltd, Rabale

A Practice School-I Station of

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI

(JULY,2022)

INDEX

Practice School Details

STATION: Prama Instruments Pvt Ltd

CENTRE: Rabale, Navi Mumbai, Maharashtra

Date of start: 31st May 2022 **Date of submission:** 21st July 2022

TITLE OF PROJECT: Single Axis Solar Tracker

ID NO.	NAME	DISCIPLINE
2020A3B50708G	Abhinav Srivastava	EEE
2020A8PS1799P	Atharva Uday Deshmukh	ENI
2020A8PS2156H	Saksham Subhash Yadav	ENI

PS INSTRUCTOR: Ms Anushaya Mohapatra

INDUSTRY MENTOR: Chittaranjan Sir, Kalpesh Patel Sir

PROJECT DOMAIN: Electronics

PROJECT AREAS: Microcontrollers, Sensors, Actuators

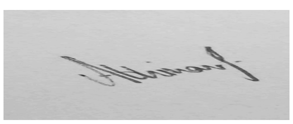
PROJECT KEYWORDS: Solar Tracker, Maximise Solar Output, Arduino,Raspberry Pi Pico, ESP8266 WiFimodule, API ,3D printing,CAD modelling

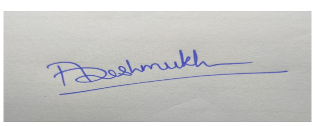
ABSTRACT OF THE PROJECT

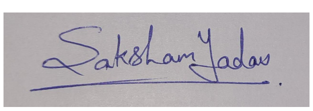
Through this project, we intended to make a solar tracker package as inexpensive as possible in addition to make it easy to install for a layman installing a Solar Panel and increase the efficiency of the Solar Panel in terms of Power generated to make it a way of being a more sustainable and more efficient way of electricity generation.

We planned to incorporate fewer components required in a package and make as many versions of the Tracker as possible with different components and functionalities for a consumer to choose from. We have a bare minimum model with only the basic formalities. Then we have the Calibration and Read model, which stores calibrated data and then uses the data for optimising the position of the Solar Panel. Then the most advanced version includes reading real-time data from a website and optimising the Solar Panel's position.

Also, we tried to make the package efficient enough to minimise maintenance costs and make it work seamlessly.







Signature of Students | **Signature of Faculty**

Date: 21/07/22 | **Date:22/07/22**

ACKNOWLEDGEMENTS

We would like to thank our PS instructor, Ms Anushaya Mohapatra Ma'am, for the smooth conduct of our Practice School Term. We would also like to thank Chittaranjan Sir for giving us this project and allowing us to work in the premises of Prama Instruments Pvt Ltd.

We would also like to thank Kalpesh Sir, and Shubhangi ma ' am for suggesting better ways of getting work done, providing solutions, and being there at every step. We would also like to thank Yogesh Sir for his technical help with the 3D printer.

Lastly, We would like to thank all the staff members at Prama who have been so welcoming and helpful and thank everyone who worked towards making the Practice School session being possible.

Introduction

Solar Panel

Solar panels are devices used to absorb the sun's rays and convert them into electricity or heat.

A solar panel is a collection of solar (or photovoltaic) cells, which can be used to generate electricity through the photovoltaic effect. These cells are arranged in a grid-like pattern on the surface of solar panels. Thus, it may also be described as a set of photovoltaic modules mounted on a supporting structure.

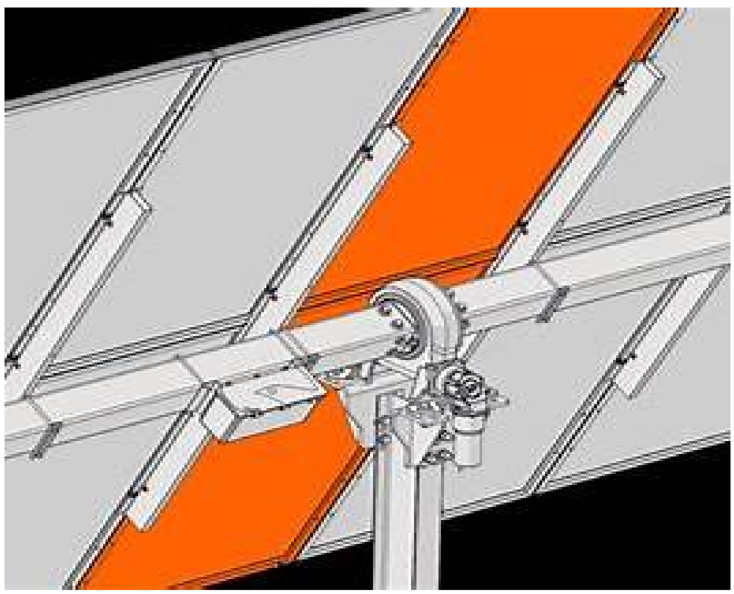
Solar Tracker

What is a solar tracker?

A solar tracker is a mechanical device - an automatic act to favourably respect the sun's rays from a photovoltaic panel, a solar thermal panel or a solar concentrator. The solar tracker causes an increase in the power of the captured solar energy and, therefore, the actual performance of the renewable energy device. One type of solar tracker is the heliostat. Historically, the first solar tracking systems were those in orbit on artificial satellites in the respective solar panels.

The main objective of a tracker is to maximise the efficiency of the device housed on board. In the photovoltaic field, modules mounted on a tracker are generally arranged geometrically on a single panel, which avoids using a tracker for each module.

In the area of solar concentration, a solar tracker is helpful to keep the fire point generated by the paraboloid constant in the channelling element of the liquid to be heated. The greater the perpendicular alignment with the sun's rays, the greater the efficiency of thermodynamic conversion and the thermal energy produced for the same surface; the lower the solar panel surface required for the same required power, the lower the installation costs.



Advantages of Solar Tracker

Trackers generate more electricity than their stationary counterparts due to increased direct exposure to solar rays. This increase can be as much as 10 to 25%, depending on the geographic location of the tracking system.

There are many solar trackers, such as single-axis and dual-axis trackers, that can be the perfect fit for a unique job site. Installation size, local weather, degree of latitude and electrical requirements are all important considerations that can influence the type of solar tracker best suited for a specific solar installation.

Solar trackers generate more electricity in roughly the same amount of space needed for fixed-tilt systems, making them ideal for optimising land usage.

In certain states, some utilities offer Time of Use (TOU) rate plans for solar power, which means the utility will purchase the power generated during the peak time of the day at a higher rate. In this case, generating a more significant amount of electricity during these peak times of the day is beneficial. A tracking system helps maximise the energy gains during these peak periods.

Advancements in technology and reliability in electronics and mechanics have drastically reduced long-term maintenance concerns for tracking systems.

Classifications of solar trackers

According to their construction characteristics, solar trackers, are divided according to degrees of freedom offered. Solar trackers can offer the panel freedom of mono or biaxial movement.power supplied to the orientation mechanism;type of electronic control.

Trackers to a Degree of Freedom

X-Axis:

Tilt trackers are the simplest to make and rotate around the east-west axis. The solar panel rises or falls (usually manually twice a year) towards the horizon so that the angle to the ground is statistically optimal according to seasonality. In practice, a tilt follower is made using telescopic mechanical profiles to raise or lower the photovoltaic panel with respect to the horizon. Conceptually similar to the liftable shelf of a school desk, these trackers offer an increase in production of less than 10%, to rarely justify a servomechanism.

Y-Axis:

The trackers aim to follow the sun throughout the sky on their daily journey, regardless of the season of use. In this case, the axis of rotation is north-south, while the height of the Sun above the horizon is ignored. These trackers are particularly suitable for low-latitude countries, where the sun's path is, on average, wider during the year. The rotation required for these structures is wider than the inclination, sometimes up to ± 60 °.

These followers make each row of photovoltaic modules look like a grill facing the equator. This type of tracker manages to have a higher performance than the trackers along the X-axis and allows to increase energy production by approximately 15%, compared to a fixed photovoltaic system.

An advanced feature of these followers is called recoil and solves the problem of shading that the rows of photovoltaic modules inevitably cause at sunrise and sunset rising to the horizon. This technique requires that the servomechanisms orient the modules according to the sun's rays only in the central band of the day but reverse tracking near sunrise and sunset.

The nocturnal position of a photovoltaic field with recoil is perfectly horizontal with respect to the ground. After dawn, the misalignment of the modules orthogonal to the sun's rays is progressively reduced as the shadows allow. A similar procedure is performed before sunset, returning the photovoltaic field to a horizontal position at night.

Z-Axis:

Azimuth (or yaw) followers have a degree of freedom with a zenith axis - nadir. To achieve this, the panel is mounted on a servo-assisted rotating base, flush with the ground. The resulting increase in electricity production is approximately 25%.

Polar Axis:

The polar axis trackers move on a single axis inclined with respect to the ground and approximately parallel to the axis of rotation of the earth. This axis is similar to the one where the sun draws its path in the sky. The axis is identical but not the same due to the variations in the height of the sun's path with respect to the ground in the different seasons. Therefore, this system of rotation of the photovoltaic panel around a single axis manages to keep the panel perpendicular to the sun throughout the day (always neglecting the summer-winter oscillations of the sun's path) and provides the maximum efficiency that can be obtained with a single axis of rotation.

Types of Solar Trackers

The sun's position in the sky varies with the seasons (elevation) and time of the day as the sun moves across the sky. Hence there are two types of Solar Tracker: Single Axis Solar Tracker: Single Axis Solar Trackers can either have a horizontal or a vertical axle. They are primarily built for tracking the sun from east to west direction. Dual Axis Solar Tracker: Double axis solar trackers have both a horizontal and vertical axle and can track the sun's motion anywhere in the world.

Single Axis Solar Tracker:

Single-axis trackers are a technology that adjusts the position of a solar panel along an axis to follow the sun’s changing position throughout the passing days and years. The panel is adapted to create the smallest angle of incidence (the angle at which the sun hits a solar panel). The trackers tilt on a singular axis to follow the sun from east to west as it moves throughout the course of each day to maximise energy production.

Solar Trackers with Two Degrees of Freedom:

The most sophisticated solar trackers have two degrees of freedom to perfectly align the orthogonal photovoltaic panels with the sun's rays in real-time. The cheapest, but not the only, way to do them is to mount one tracker on another. With these solar trackers, there are increases in electricity production that reach up to 35 - 40%, but with greater constructive complexity.

Power Supply

Based on the power necessary for the persecutor's movement, we can divide them into

Active trackers, if activated by gears or motors;

Passive trackers, if they are set in motion by autonomous physical phenomena, such as the thermal expansion of gas or whatever.

Methodology

After the project was allotted to us, we began brainstorming on the parameters provided to us by our Industry Mentor, Kalpesh Sir. The parameters primarily being cost-effective, easy to install, factory production-ready and optimised for maximum efficiency.

The first step was to identify the outputs required from the final component.

Outputs Required:-

- Sensing the passage of the Sun overhead.
- Changing the inclination of the solar panel accordingly
- Able to store the said changes in retaining that information
- Able to map the said changes with time
- Track the movement of the solar tracker with respect to the original position
- Ability to function without the live tracking of Sun
- After analysing the outputs required we began to identify the components which would help in achieving said outputs and yet operated within the parameters.
- A microcontroller was needed to take and process the inputs and thereby execute the outputs.

Input Devices:

- LDR sensors to measure the intensity and identify the position of the Sun
- MPU 6050 module to measure the inclination of the Solar Panel in the form of a gyroscope.
- RTC module acts as a Real-Time Clock which would retain the passage of time even after the microcontroller is switched off and conserve energy.
- EEPROM 24C04 acts as a memory storage unit that will store the data and retain it even after the device is shut down.

Output Devices:

- Motor Drivers, this acts as a medium between the microcontroller and the motor which drives the solar panel
- Linear Actuator, this has linear motion in one direction which will be used to push the solar panel.

Microcontroller:

We used the Arduino Nano initially to execute the codes as we were more familiar with it. The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328. It has more or less the same functionality as the Arduino. It lacks only a DC power jack and works with a Mini-B USB cable instead of a standard one. After completing a basic prototype on Arduino Nano, Sir asked us to up the cost efficiency by incorporating the code on the much better, novel and cheaper microcontroller which was released in January 2021. None of us had any prior experience in working with the Raspberry Pi Pico microcontroller, but with its better performance and industry-friendly price, we had to use it instead.

Raspberry Pi Pico's features include but are not limited to

- RP2040 microcontroller chip designed by Raspberry Pi in the United Kingdom
- Dual-core Arm Cortex M0+ processor, flexible clock running up to 133 MHz
- 264kB of SRAM, and 2MB of on-board Flash memory
- Castellated module allows soldering direct to carrier boards
- USB 1.1 with device and host support
- Low-power sleep and dormant modes
- Drag-and-drop programming using mass storage over USB
- 26 × multi-function GPIO pins
- 2 × SPI, 2 × I2C, 2 × UART, 3 × 12-bit ADC, 16 × controllable PWM channels
- Accurate clock and timer on-chip
- Temperature sensor
- Accelerated floating-point libraries on-chip
- 8 × Programmable I/O (PIO) state machines for custom peripheral support

A direct comparison between Raspberry Pi PIco and Arduino Nano is given

Specifications	**Raspberry Pi Pico**	**Arduino Nano**
Microcontroller	RP2040	ATMega4809
Size	21mm × 51mm	45mm x 18mm
Processing Speed	133 MHz	20MHz
Memory	264KB	48KB
Power Input	1.8V–5.5V	5V - 21V
Connectors	2x UART, 2x SPI, 2xI2C, 16x PWM, 3x Analog	1x UART, 1x SPI, 1xI2C, 5x PWM, 8x Analog
Price	$4	$10

As seen from the comparison we can concur why Pi Pico is better and well suited over Arduino Nano:

- Better Processing Speed(138 MHz Vs 20 MHz)
- More Memory (264 kb Vs 48 kb)
- More Input/Output Devices (2x UART, 2x SPI, 2xI2C, 16x PWM, 3x AnalogVs 1x UART, 1x SPI, 1xI2C, 5x PWM, 8x Analog)
- Cheaper(₹349 Vs ₹759)
- Unavailability of Arduino Nano in Semiconductor Market

ESP8266-Wifi Module

ESP8266 is a low-cost WiFi module that belongs to ESP's family which you can use to control your electronics projects anywhere in the world. It has an in-built microcontroller and a 1MB flash allowing it to connect to WiFi. The TCP/IP protocol stack allows the module to communicate with WiFi signals. The maximum working voltage of the module is 3.3v so you cant supply 5v as it will fry the module.

ESP8266 Features:

- Processor: L106 32-bit RISC microprocessor core based on the Tensilica Diamond Standard 106Micro running at 80 MHz
- Memory:
  - 32 KiB instruction RAM
  - 32 KiB instruction cache RAM
  - 80 KiB user-data RAM
  - 16 KiB ETS system-data RAM
- External QSPI flash: up to 16 MiB is supported (512 KiB to 4 MiB typically included)
- IEEE 802.11 b/g/n Wi-Fi
  - Integrated TR switch, balun, LNA, power amplifier and matching network
  - WEP or WPA/WPA2 authentication, or open networks
- 17 GPIO pins
- Serial Peripheral Interface Bus (SPI)
- I²C (software implementation)[
- I²S interfaces with DMA (sharing pins with GPIO)
- UART on dedicated pins, plus a transmit-only UART can be enabled on GPIO2
- 10-bit ADC (successive approximation ADC)

Components Used:

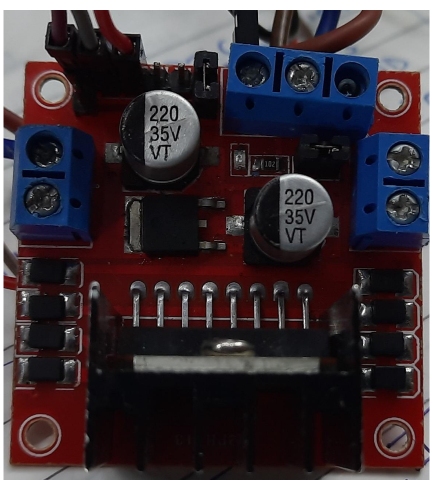
Linear Actuator

Used to change the angle at which the Solar Panel faces the Sun.

This electric push rod is a kind of electric driving device which transforms the rotary motion of the motor into the linear reciprocating motion of the pushrod.

Motor Driver L298N

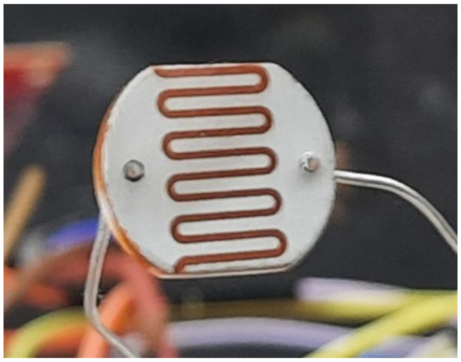
It is used to control the movement of the Linear Actuator.



L298N 2A Based Motor Driver is a high-power motor driver perfect for driving DC Motors and Stepper Motors.

Light Dependent Resistor

Used to find the intensity of sunlight falling on the Solar Panel.

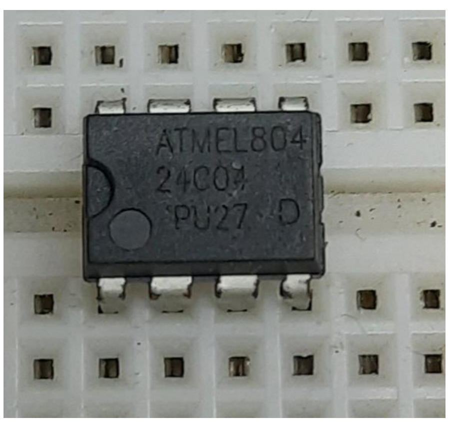


The resistance of 20mm GL20528 Light Sensitive Photoresistor LDR changes with the change in the ambient light exposure on the surface of the sensor.

As the light on the sensor increases then the resistance across the two leads decreases.

24C04 EEPROM

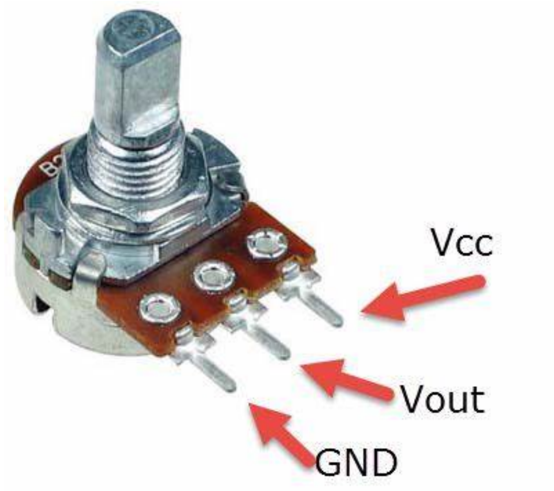
To save data from Arduino and keep the data safe even when the switch is off.



If you need to do some data storage in Arduino, then using the EEPROM is probably the most simple practice.

Potentiometer

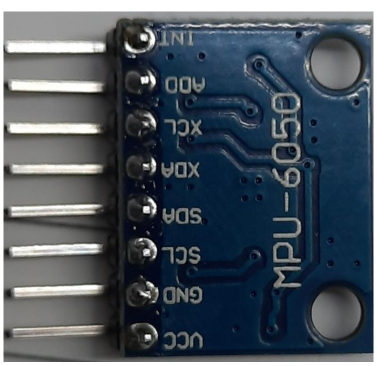
To find the potentiometer reading for controlling linear actuator.

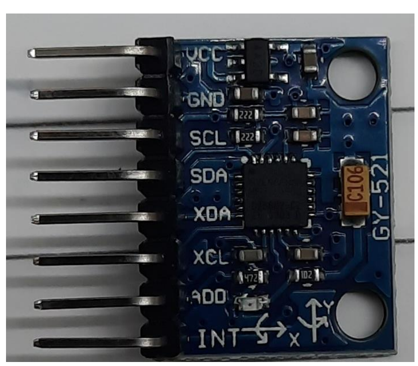


A potentiometer (also known as a pot or potmeter) is defined as a 3 terminal variable resistor in which the resistance is manually varied to control the flow of electric current. A potentiometer acts as an adjustable voltage divider.

MPU-6050 [3-Axis Accelerometer and Gyro Sensor]

Used to measure the inclination of the panel with respect to the horizontal plane.

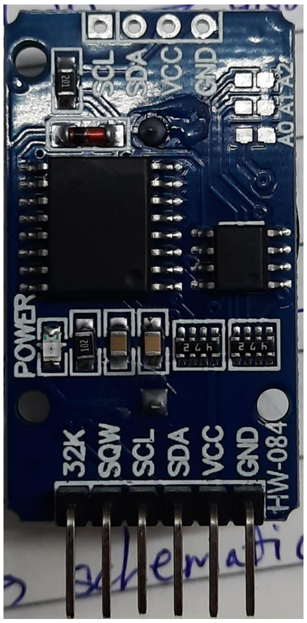


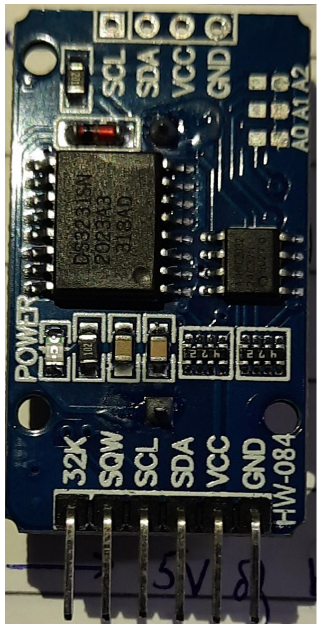


The MPU6050 devices combine a 3-axis gyroscope and a 3-axis accelerometer on the same silicon together with an onboard Digital Motion Processor (DMP) capable of processing complex 9-axis MotionFusion algorithms.

DS3231 RTC chip(Real-Time Clock Module)

Used to have an external clock to synchronise time over all components even when the microcontroller is switched off.

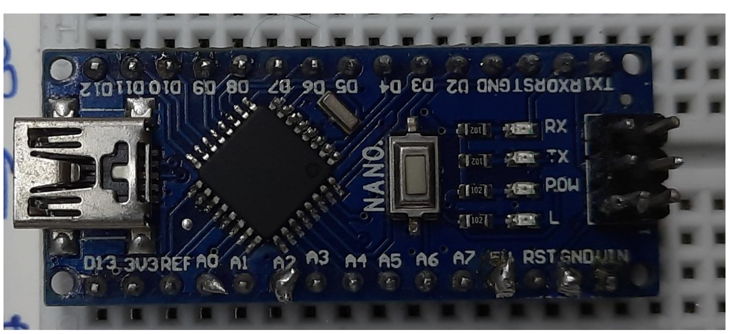




A real-time clock (RTC) is an IC that keeps an updated track of the current time. This information can be read by a microprocessor, usually over a serial interface to facilitate the software performing functions that are time-dependent.

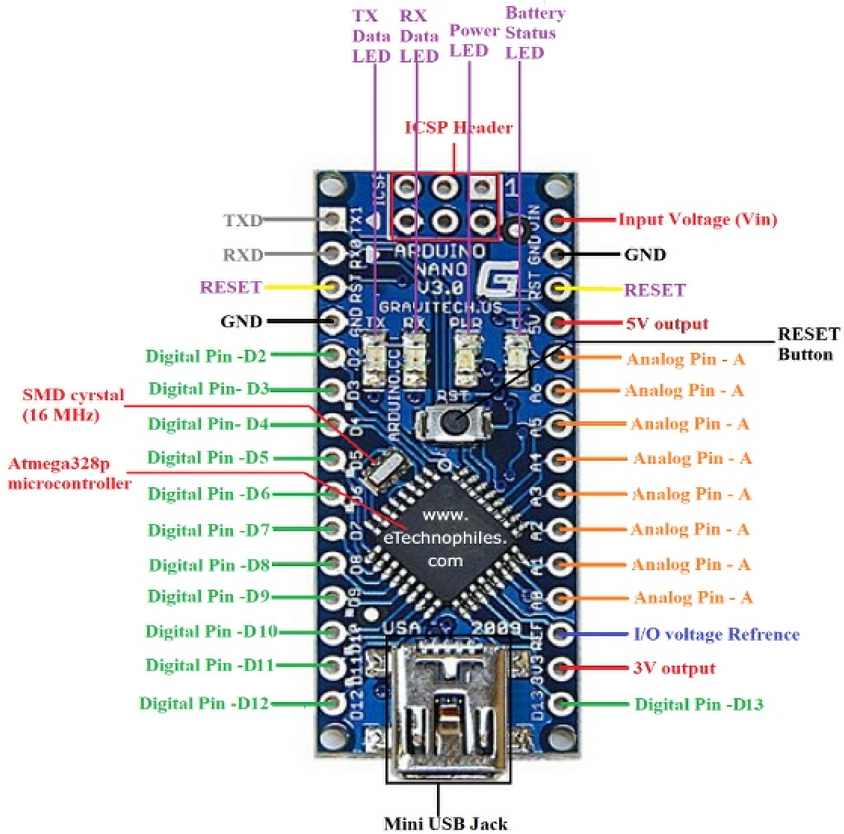
Arduino Nano

A microcontroller is used to process data given and give appropriate output.



Arduino Nano is a surface mount breadboard embedded version with integrated USB. It is the smallest, complete, and breadboard-friendly.

Arduino nano pinout:

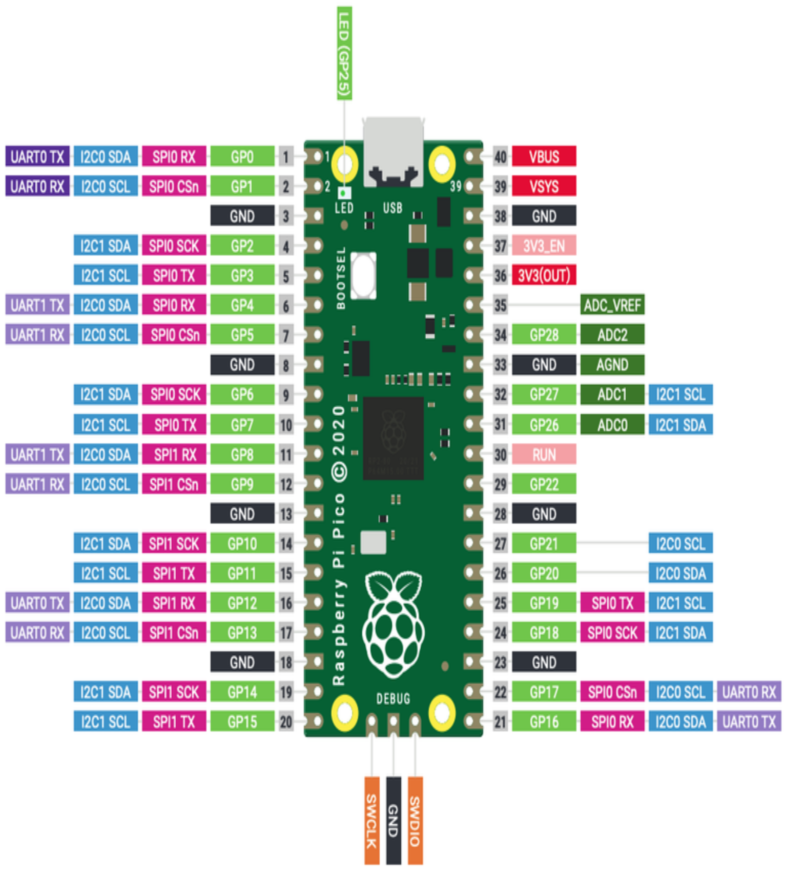


Raspberry Pi Pico

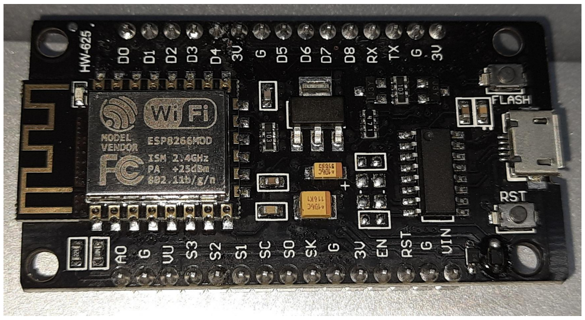
A microcontroller is used to process data given and give appropriate output.

It is equipped with an RP2040 Microcontroller chip developed by Raspberry Pi Foundation itself. RP2040 is their first dual-core ARM Cortex M0+ processor-based latest small-sized, budget-friendly microcontroller.

Raspberry Pi Pico pinout:

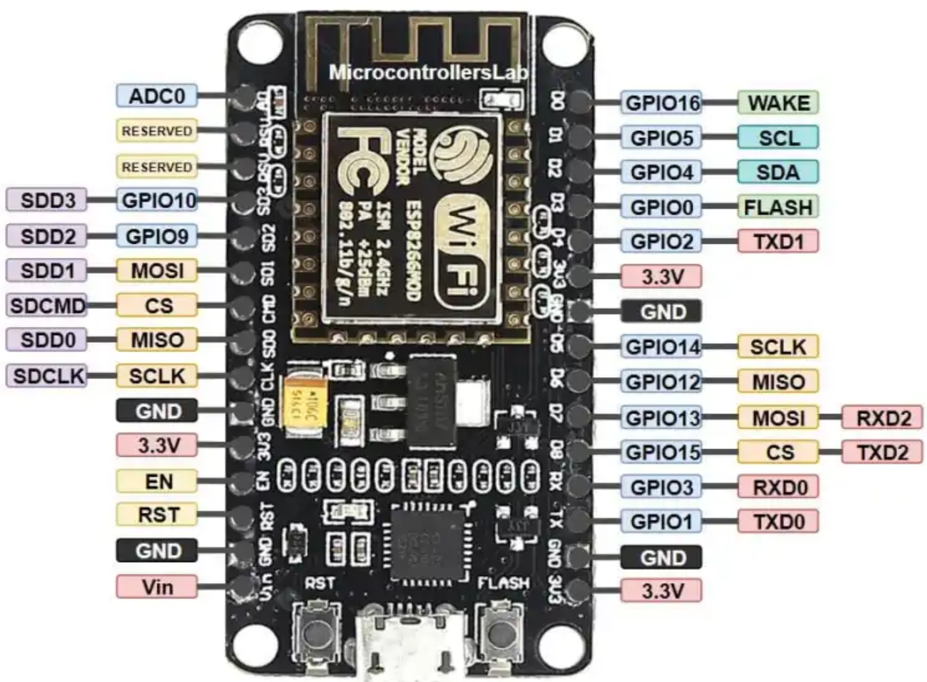


ESP 8266 with Wifi Module

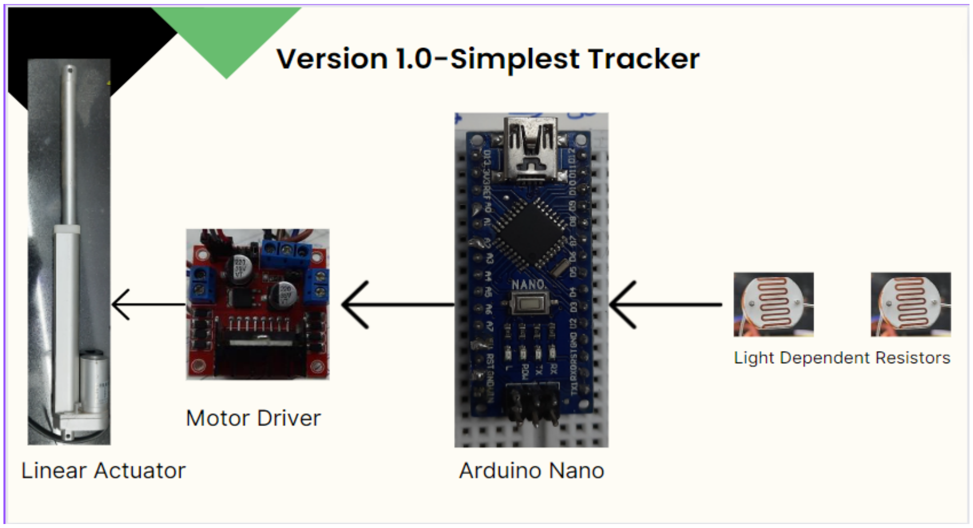


The ESP8266 NodeMCU CP2102 board has ESP8266 which is a highly integrated chip designed for the needs of a new connected world. It offers a complete and self-contained Wi-Fi networking solution, allowing it to either host the application or to offload all Wi-Fi networking functions from another application processor.

ESP8266 pinout:

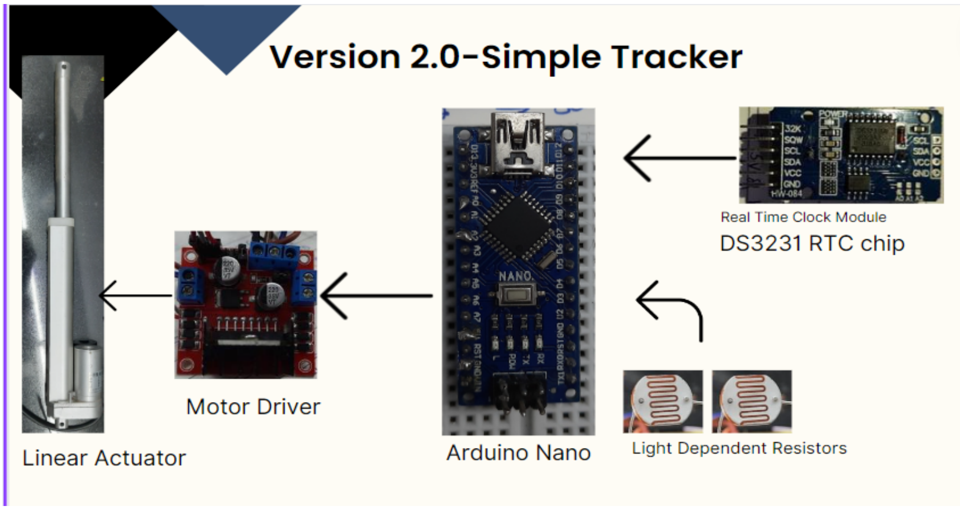


Working



After incorporating all the components we arrived at the following flowchart of information:

In the basic model of the solar tracker, the two LDR measures the difference between the amount of light falling on each sensor, on the basis of which the microcontroller decides the direction and the movement of the linear actuator in order to optimise the amount of solar radiation received by the Solar panels.

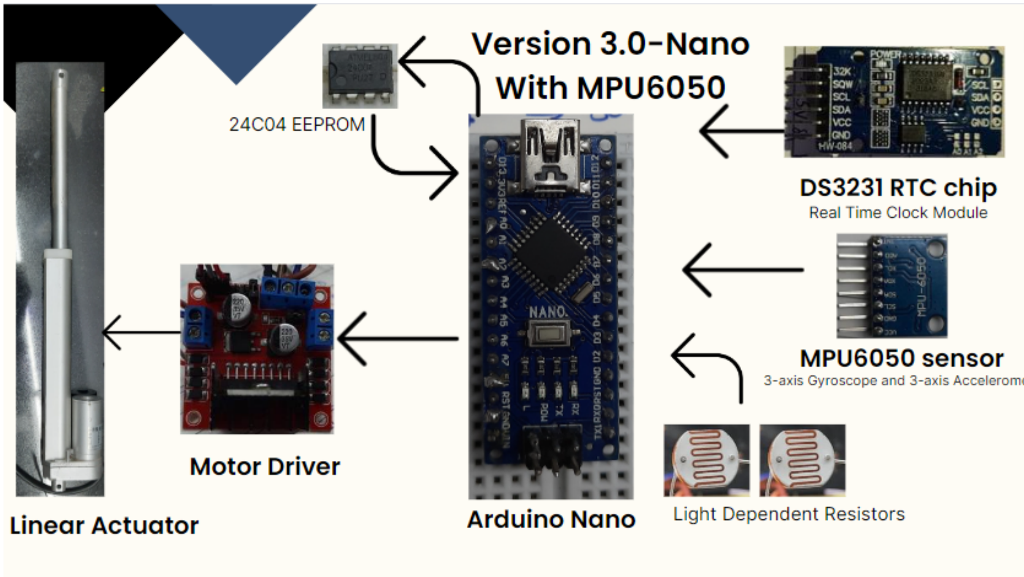


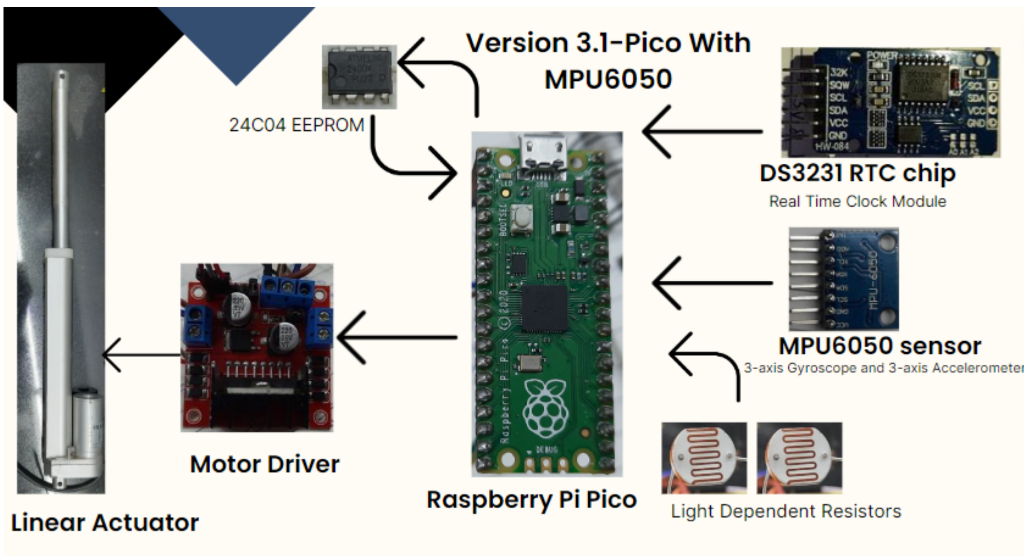
Function of RTC

Sleep Mode Initiation along with Homing

- We all know that Solar panels cannot generate electricity after sunset.
- Therefore, to avoid optimization of the actuator during night time is made sure by the Real Time Clock Module which halts the work of the tracker during the specified night hours.
- The Panel is homed when the night hour starts,i.e. the panel is brought in a resting position to be kept in night.

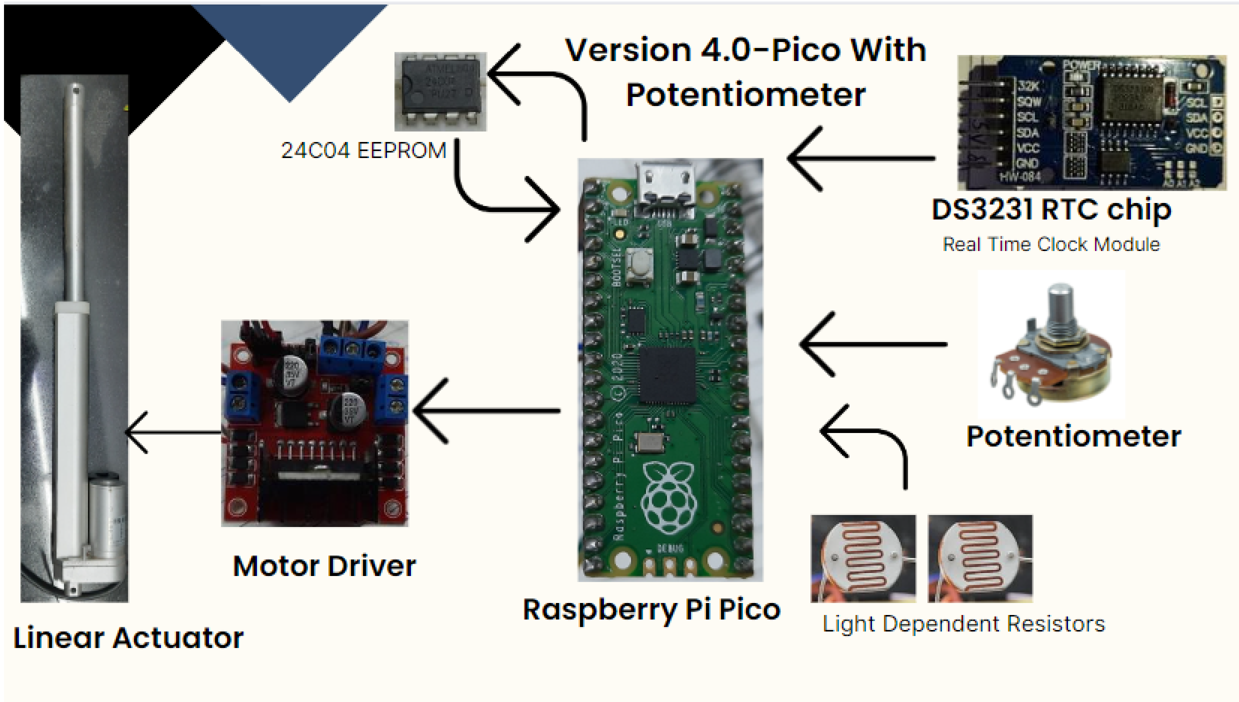
This functionality can be added in all the version of the Solar tracker to avoid wastage of power in optimising the position of Solar Panel.





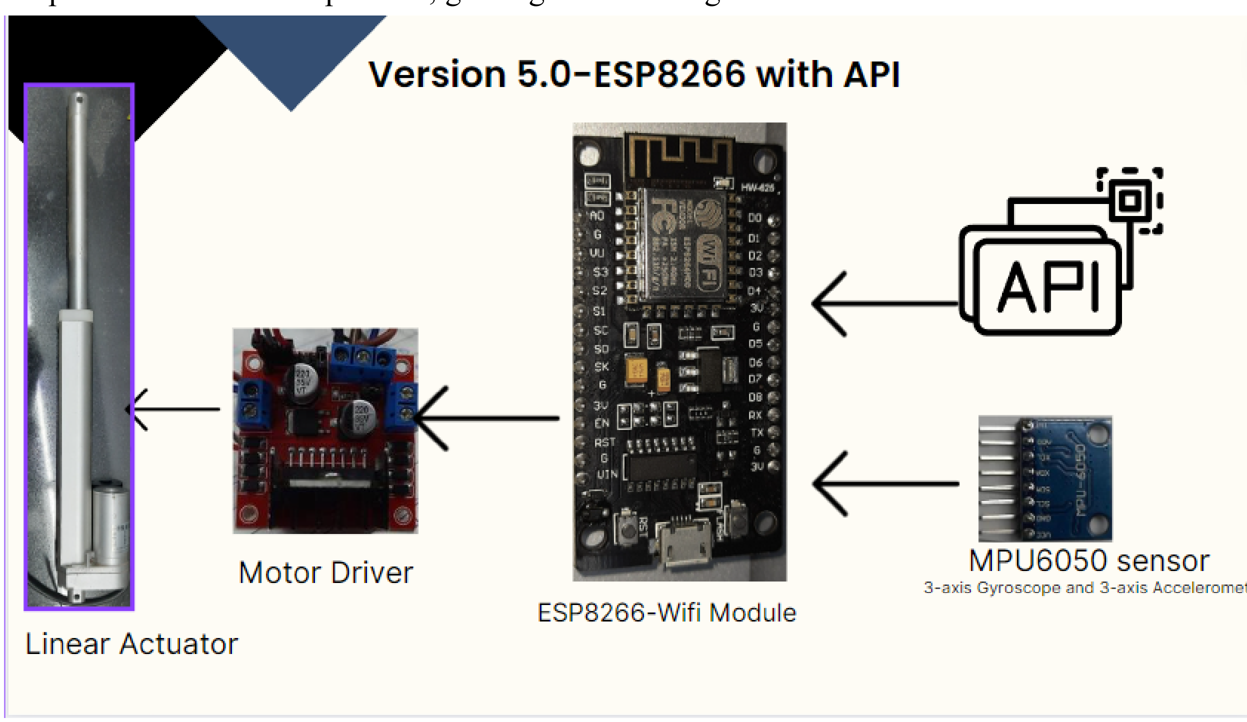
For Version 3.0 and 3.1, in Calibration Mode, the change made in the reading of MPU6050 is stored in EEPROM, when the linear actuator moves according to the difference found in LDR readings.

For Daily Mode, the reading along with the time is read and the linear actuator is commanded to optimize Solar Panel's position,ignoring LDR readings.



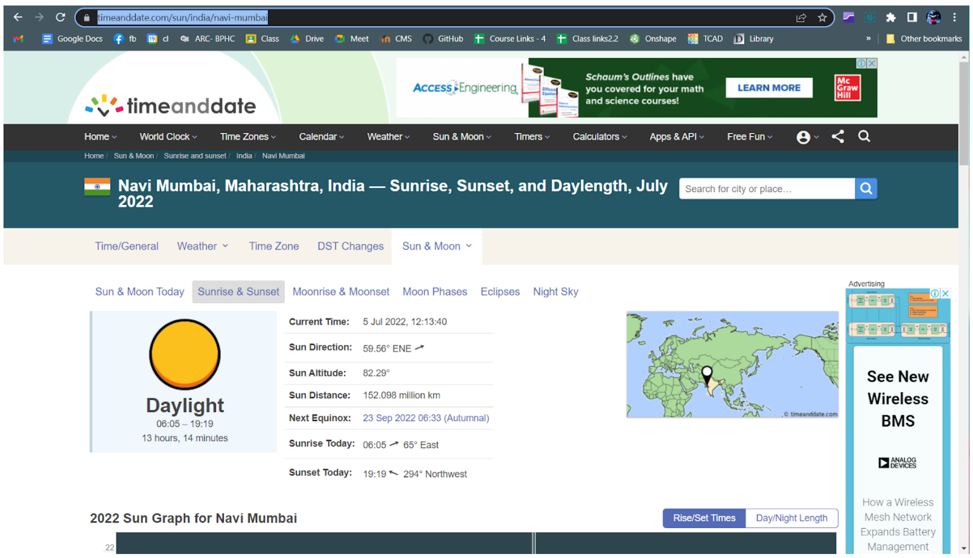
For Version 4.0, in Calibration Mode, the change made in the reading of potentiometer is stored in EEPROM, when the linear actuator moves according to the difference found in LDR readings.

For Daily Mode, the reading along with the time is read and the linear actuator is commanded to optimize Solar Panel's position,ignoring LDR readings.

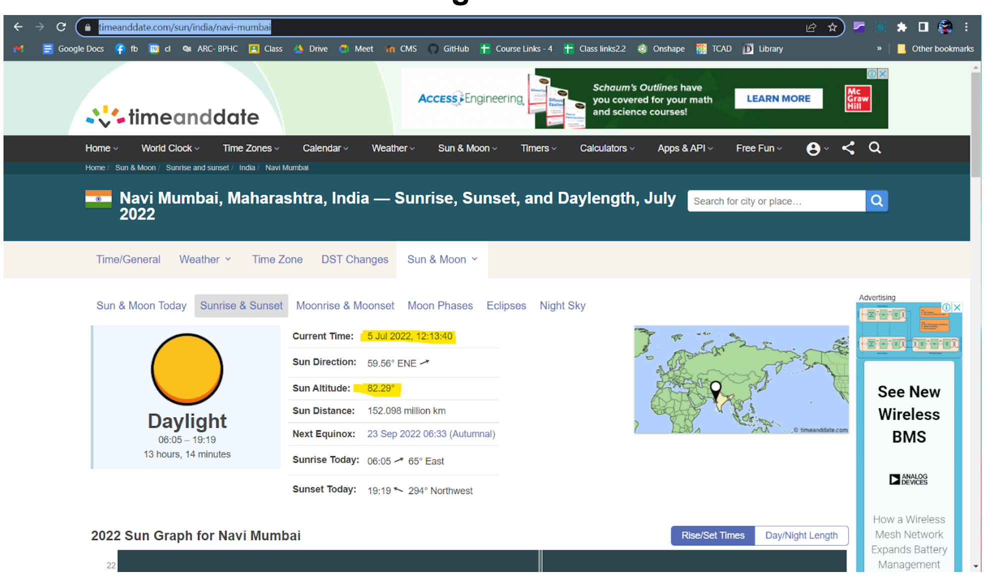


In this Version, We take data from a weather website i.e. timeanddate.com in real-time.

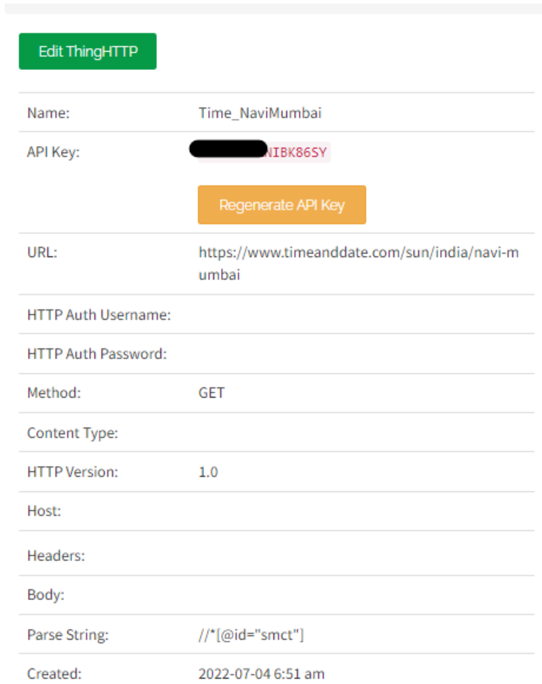
Website for data to be taken: [Solar Data of Navi Mumbai]

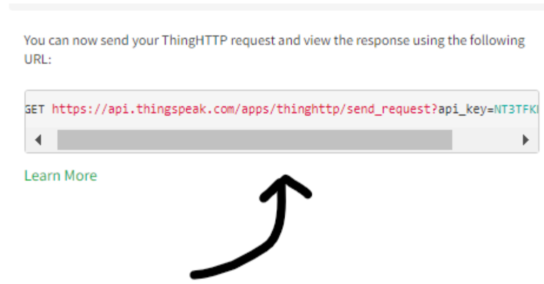


Relevant Data Marked for Usage:



Used [Things Speak]() for API creation





Now the API can be used through the given URL.

The next step would be to incorporate the URL in the code in the Arduino IDE,that is well documented in the github.

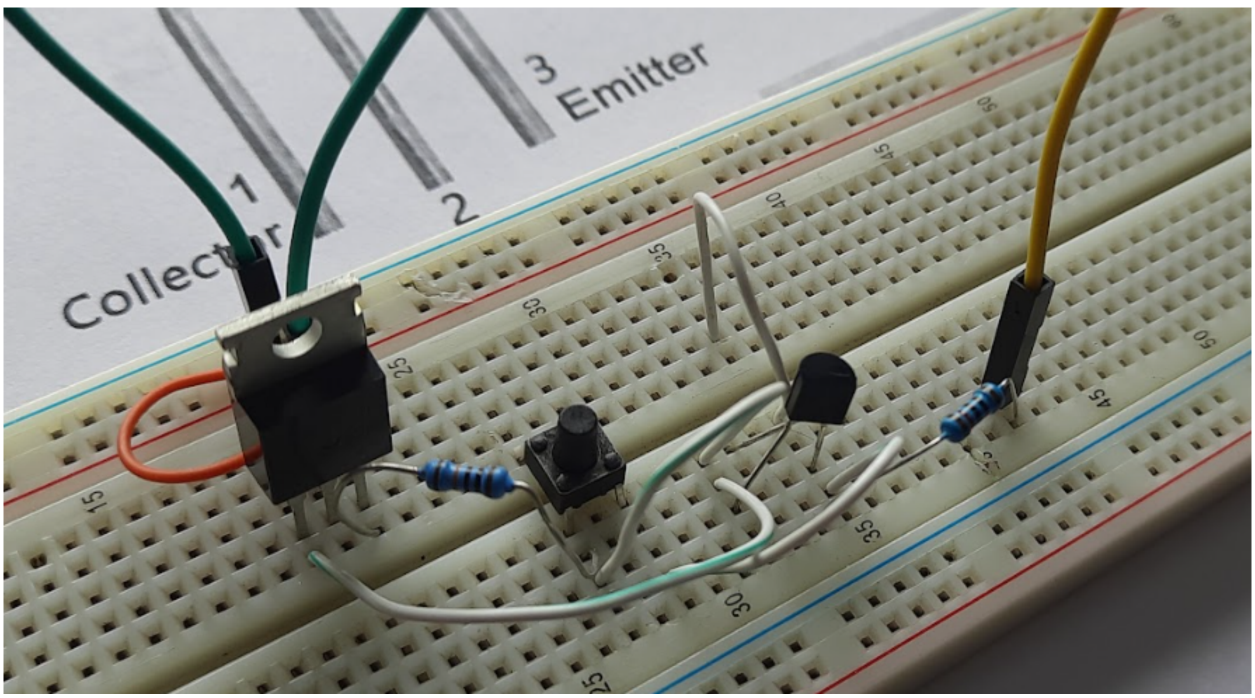
A step by step tutorial on how to make an API is also present in the github as well.

Timeline Of the Project

Our work on PS onsite started by familiarising us with the essential electronic components used in the industry.

The first task allotted to us was constructing a Soft Switch commonly used in the electronics industry. We were given a schematic and components listed in the schematic. We had to brainstorm the errors in the schematic and tweak them to produce the results.

We changed the schematic and introduced the capacitors and a couple of diodes into the circuit. This made the circuit compatible with the software (Arduino IDE).



After completing the first task and getting familiarised with the resources available, we brainstormed ideas for the PS project. After running through several hobby ideas, Sir allotted us the Single Axis Solar Tracker project as it was an Industry level project where optimization and cost management were the key features to work on, apart from the basic schematic and coding involved.

Some time was spent analysing the prototype already developed in Prama Instruments. Dismantling the prototype gave us an idea of the components used and areas for improvement.

After planning the schematic and developing a BOM(Bill of Materials), we worked on the project's code.

Here is the link to the BOM we submitted.[BOM]

We wrote code for the difference between LDR readings and accordingly controlled the linear actuator. Another code was written to find angles from MPU 6050, then we were able to use the RTC chip and we used python for serial communication and matplotlib to plot data from the potentiometer. We had the code ready in segments. We needed to incorporate them in one sketch.

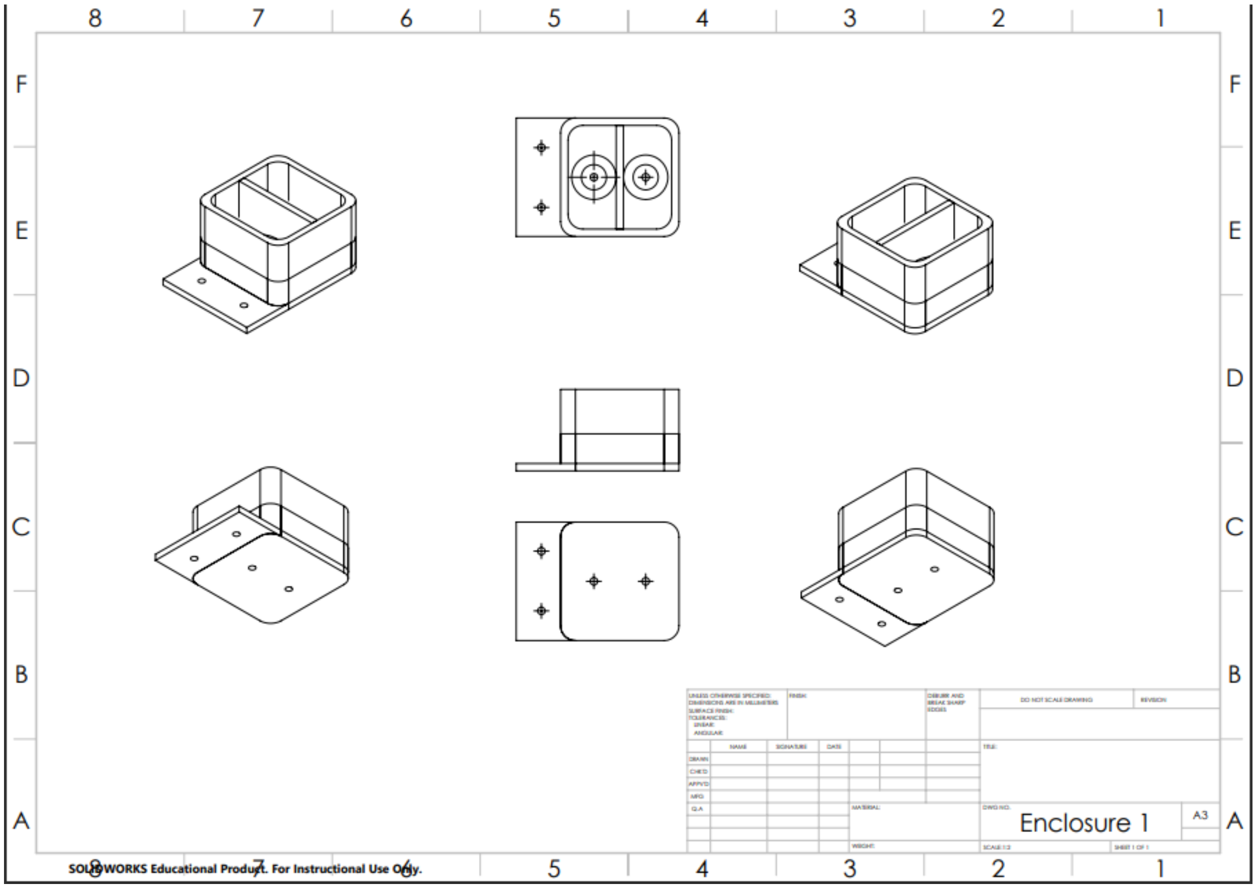
After incorporating LDRs and the Linear Actuator in one code we moved on to integrating the RTC module with the code but ran into some problems. It was an input-output problem which was worked around and sorted.

We were then able to combine the LDRs, Actuator and the RTC module into one code and our draft 5 was ready.

Here is a link to one of the demonstration videos.[Demo Video]

For making the solar tracker even more concise and cost-effective Sir suggested the device 'EEPROM' to store the data which will be permanent and can be retained even after the whole appliance shuts down to save power.

Our next step involved making the first draft of the CAD model for the enclosure for the LDR sensor as they need to be protected from the exterior.



We took data at the terrace from LDR directly and through a made-up enclosure made of cardboard which inspired our cad model. This was done to get an idea of the range of data we need to program the code for.

EEPROM is a novel device which none of us had worked on, researched and figured out how to use the EEPROM of ARDUINO MEGA and successfully stored the LDR difference value every 20 seconds (the time interval can be changed as per our convenience).

First, we worked on programming the EEPROM of the microcontroller Arduino Mega given to us. After that, we moved to work on programming external EEPROMs as we would not be using Arduino Mega in the final schematic, as its features are very far-fetched from our needs.

We had to use i2c communication for using an EEPROM and therefore needed external EEPROM as there was no option to update the data, and only writing was possible, hence we couldn't risk writing to the EEPROM of the microcontroller.

Sir suggested we change the base microcontroller to a novel and powerful microcontroller Raspberry Pi Pico, it is a much more evolved microcontroller, and its features will be discussed further.

We installed the Raspberry Pi Pico environment to Arduino IDE and started running our sample codes on it.

We tested all the code that we worked on with Nano on Pico except the EEPROM part.

We also tested the MPU6050 module through Nano. We were able to get the output from it but we needed to optimise that output to suit our needs.

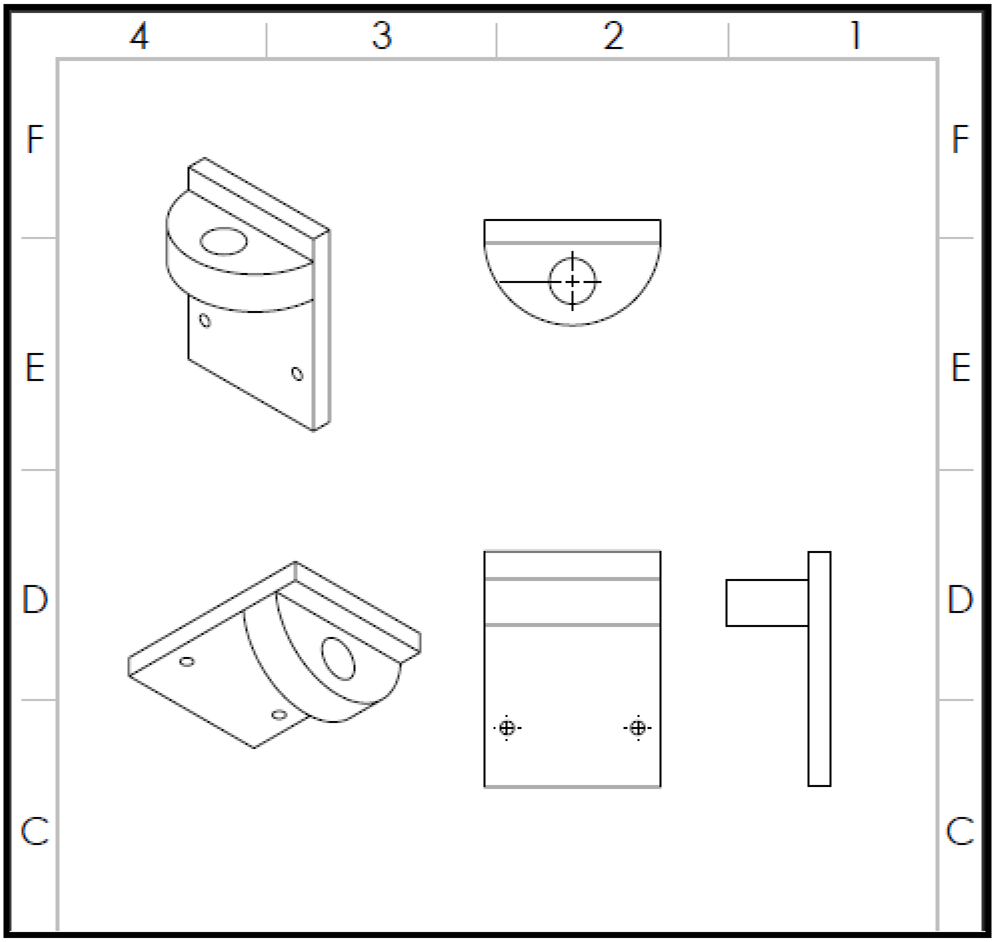
After analysing the output from the MPU 6050 we were able to jot down a code for obtaining the relevant data in the form we needed. Primarily we were getting 3 sets of data in Yaw, Pitch and Roll. We calibrated the sensor such that the data we received was not universal but rather set to a 0 on initialization and changed accordingly with further movement. We also reduced the data set to 2 data: Yaw and Pitch.

Having tested out each component individually our step included incorporating all the components together in an optimised circuit diagram. We first made a circuit diagram using software tools to reduce the time in desoldering. Having done that we used a PCB to solder the components and wires onto it.

After all the components were assembled and soldered onto the PCB we began testing. During testing we ordered several issues such as shortening of the circuit and dry solders, we received guidance on the same from Shubhangi Ma'am. Having rectified most of the errors we proceeded forward.

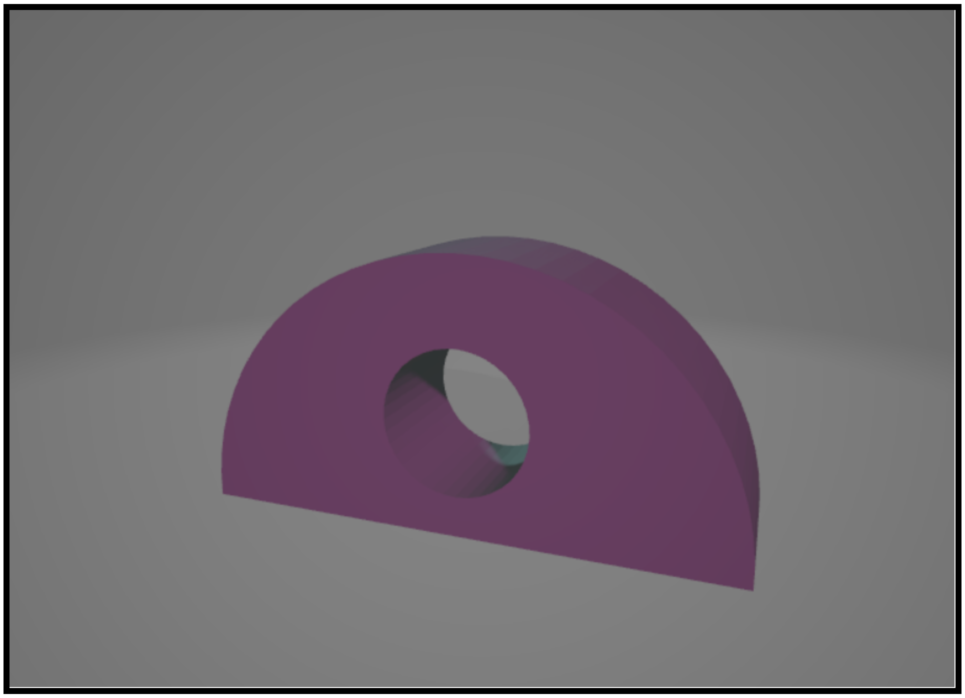
A major setback occurred during this process, our MPU6050 was damaged and not working properly. Due to which we couldn't proceed ahead. Ordering of the new component would take a lot of time. We contacted Sir for a resolution, he suggested we incorporate a Potentiometer, this was a common practice in industry products to use a potentiometer to gauge the movement and keep a measure of the same in electronic moving parts.

We set out to design an attachment which would move the potentiometer along the entire assembly. We had to incorporate the movement of the linear actuator with the movement of the potentiometer. The way we designed it was to use the linear motion of the linear actuator which produces the angular motion in the solar panel to in turn give us the desired revolutionary motion in the potentiometer component.



This attachment would be mounted onto the frame itself.

We prepared a 3D printable file in the STL format for the same. The curved part would be 3D printed and the rest would be a Mild Steel Plate welded on the frame. The 3D printed part would be stuck onto the plate. We chose 3D as it would be much easier to print a complex geometry and plastic-based part would be easier to manage than metal.



During the fabrication process, we were also working on the third method to have a functional single-axis solar panel. This method uses ESP8266 WiFimodule, a WiFi module that can receive and send data over WiFi which is the next step into home automation. Hence our product can be controlled using any WiFi-enabled device, essentially a smartphone where the user will not have to go through the calibration or sensor-based phases.

We worked on developing a code such that data could be extracted from the website which provides RealTime inclination of the Sun to the ground of that particular location. This data was then used to give the output to the linear actuator to match the supposed reading of the MPU module.

With this all three methods of using a single-axis solar panel were complete.

We began with 3D printing the desired attachments and enclosures. Some time was spent on learning the functioning of the 3D printer which included Auto Levelling the bed by adjusting the desired screws, Replacing the filament in the Printer and Heating the bed and the nozzle.

During this phase, we ran into the issue of having jammed the nozzle due to incorrect heating instruction input. This has delayed the frame fabrication process.

Setbacks

We experienced 2 major setbacks:

- **MPU6050 Malfunction**: We had to change the entire approach by incorporating a Potentiometer instead and therefore plan the necessary frame changes.
- **3D printer motor/nozzle Jam:** This caused halting of the manufacturing process.

Results

- We developed 5 Versions of Single Axis Solar Trackers with the help of 3 Microcontrollers available to us. We incorporated different components to produce a variety of products to suit different needs based on budget and accessibility.
- All the codes have been verified and documented for further development and integration with manufactured parts.
- Manufacturing of the 4.0 version was underway but was halted due to the jamming of the 3D printer nozzle.
- File parts have been shared and can be printed when the needed servicing is done

Learnings

- Working and coding in Arduino IDE software. All the versions were developed by setting up the particular environment for respective Microcontrollers.
- Hand developing circuits and then implementing them using a breadboard, jumper cables and components.
- Soldering wires and components to the PCB to make the circuits concise to work with(common industry practice).
- Working with a 3D printer, by developing and designing components in STL format and then physically printing them on the printer after doing the needed calibration.
- Working in a structured environment and as members of a team at a formal workspace.

Conclusion

In conclusion, this Practice School session helped us to receive hands-on experience of working in the industry. In the process, we explored a lot of electronic components and the libraries associated with them.

We also got to know about incorporating components in addition to learning the usual process that goes behind choosing components for a project in the industry.

We also learnt about collaborative working and working as a team and we understood the whole process through which work commenced in the R&D department of a company like Prama Instruments Pvt Ltd.

We are thankful for the opportunity given to us from Prama Instruments as well as BITS Pilani PSD.

Documentation Links

All the Arduino code and python codes are stored in a GitHub repository, and here is the link to the repository.

[sakshamssy/PS1_PramaInstruments]

And all the schematics and demonstration videos, and images are stored in this google drive folder.

[PS1_PramaInstruments]